SPECIAL REPORT 91-11

(2)

CRREL

Operators Manual for Determining Mole Percent Purity Using IMPURE

Dean Pidgeon and Patrick B. Black

August 1991

For conversion of SI metric units to U.S./British customary units of measurement consult ASTM Standard E380, Metric Practice Guide, *published by the American Society for Testing and Materials, 1916 Race St., Philadelphia, Pa. 19103.*

This report is printed on paper that contains a minimum of 50% recycled material.

Special Report 91-11

U.S. Army Corps of Engineers
Cold Regions Research & Engineering Laboratory

Operators Manual for Determining Mole Percent Purity Using IMPURE

Dean Pidgeon and Patrick B. Black

August 1991

Prepared for
U.S. ARMY TOXIC AND HAZARDOUS MATERIALS AGENCY
CETHA-TE-CR-91043

PREFACE

This report was prepared by Dean Pidgeon, Research Technician, Geochemical Sciences Branch, Research Division, and Dr. Patrick Black, Soil Physicist, Applied Research Branch, Experimental Engineering Division, U.S. Army Cold Regions Research and Engineering Laboratory. Funding for this work was provided by the U.S. Army Toxic and Hazardous Materials Agency, Aberdeen Proving Ground, Maryland (R-90 Multi-Analytical Services), M. Stutz, Project Monitor.

The authors acknowledge the helpful suggestions of the technical reviewers, Dr. Charles Reynolds and Lawrence Perry, as well as the technical assistance of Dr. Thomas Jenkins and Marianne Walsh.

CONTENTS

Preface ii
Introduction 1
Implementation of van't Hoff's equation 1
Differential scanning calorimetry 3
DSC calibration and operation 3
Purity determination from DSC data 5
IMPURE 5
Preparation of sample 5
DSC calibration 5
DSC setup 6
Using the ASYST program IMPURE.DMO 6
Data analysis 6
Literature cited 9
Appendix A: Listing of the program IMPURE 11
Abstract 35

ILLUSTRATIONS

Figure

1. Melting curve from DSC showing eutectic melting of an impurity 2
2. Fractions curve corrected for lost area 2
3. Fractions curve not corrected for lost area 2
4. Typical melting temperature determination for indium 4
5. Baseline determination of a melting curve 4
6. Sample plotting output from IMPURE 8

TABLES

Table

1. Optimum sample parameters determined by Hunter and Blaine (1984) 5
2. Molecular weights and starting and ending temperatures for seven munition standards needed for IMPURE 6
3. Sample summary output from IMPURE 7

Operators Manual for Determining Mole Percent Purity Using IMPURE

DEAN PIDGEON AND PATRICK B. BLACK

INTRODUCTION

This report describes the procedures necessary to determine the mole percent purity of the Standard Analytical Reference Materials (SARMs) suitable for analysis by differential scanning calorimetry (DSC). The sampling method, data acquisition procedure and statistical analysis used in this work are adaptations of standard methodologies found in the literature. As there already exists abundant literature on the implementation of thermal techniques for the determination of purity, our objective is to present a simple package—IMPURE—that should be useful to the quality control community.

This report first presents a detailed description of the required interpretation of the data needed to apply van't Hoff's equation for determining molar purity. Also given is a discussion of differential scanning calorimetry that emphasizes the necessary instrument calibration to measure heat of fusion and melting point temperature. Finally the recommended laboratory procedures for purity determination are presented.

While this report is primarily concerned with the SARMs, it should be equally applicable to other crystalline organic compounds that

- Have a molar purity greater than 98 mol %;
- Have impurities that are both insoluble in the solid and soluble in the melt (i.e. they do not form solid solutions);
- Do not decompose near their melting temperatures; and
- Do not occur in multiple crystal forms.

IMPLEMENTATION OF VAN'T HOFF'S EQUATION

When a substance is heated, the entire impurity is assumed to melt at the eutectic temperature θ_E, with the remaining solid phase composed only of pure substance (Fig. 1). As the temperature increases from the eutectic temperature to the melting temperature of the substance θ_F, the mole fraction of the impurity in the liquid phase X_2 is constantly lowered as the pure substance melts (Widmann and Sommerauer 1988). The mole fraction of impurity X_2 in a melted fraction F of the major component is simply

$$X_2 = X_{2*}\ 1/F \tag{1}$$

where X_{2*} is the total mole fraction of impurity in the original substance. The influence of a small amount of impurity on the change of the melting temperature of a pure compound from θ_0 to θ_F is described by van't Hoff's equation:

$$\frac{\theta_0 - \theta_F}{X_2} = R\ \frac{\theta_{02}}{\Delta L} \tag{2}$$

where θ_0 is the melting temperature of a pure compound and θ_F is the melting temperature resulting from the inclusion of X_2 amount of impurity. R (cal mol^{-1} deg^{-1}) is the gas constant, and ΔL (cal mol^{-1}) is the latent heat of fusion of the pure compound. Substituting eq 2 into eq 1 gives a linear equation relating the fraction melted F_i to the temperature θ_{Fi}:

$$\theta_{Fi} = \theta_0 = \frac{X_{2*} R_0}{\Delta L}\ \frac{1}{F_i}\ . \tag{3}$$

A plot of inverse fraction vs temperature should be linear, with the intercept being the melting point temperature of the pure major component (Fig. 2). It is usually found, though, that the plot is concave (Fig. 3). Several rationalizations for this deviation from nonlinearity exist (Perkin-Elmer n.d., Plato and Glasgow 1969, Wiedeman et al. 1984). The simplest scenario is the "missing area" argument.

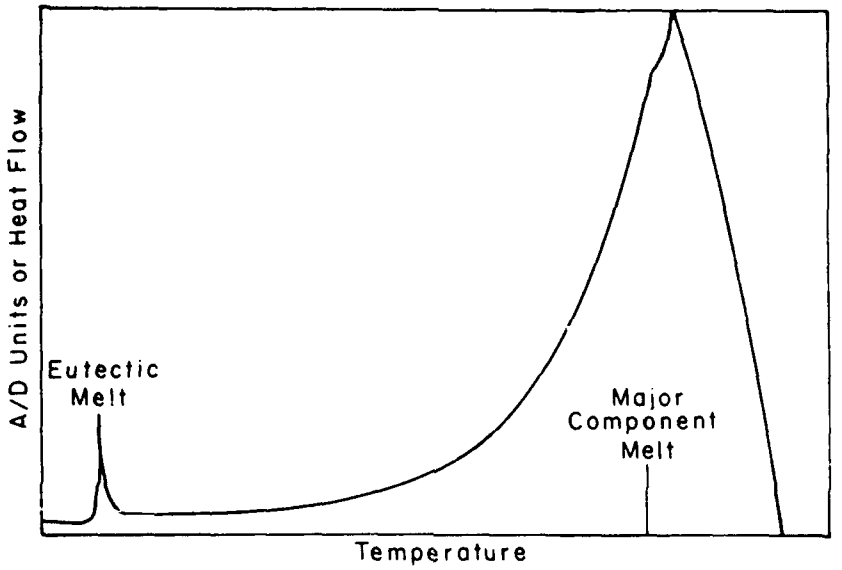

Figure 1. Melting curve from DSC showing eutectic melting of an impurity.

Since melting curve data are commonly collected at temperatures above the eutectic temperature ($\theta > \theta_E$), there is a constant amount of heat omitted. This heat must be added to the measured heat to correctly apply eq 3 (Widmann and Sommerauer 1988). As demonstrated below, the amount of heat, on a mass basis, is directly proportional to the area beneath the melting curve, so heat and area are interchangeable. The fractions are then calculated based on this additional area K:

$$\frac{1}{F_i^*} = \frac{A_{tot} + K}{A_i + K} \tag{4}$$

where A_{tot} and A_i are the total and partial areas beneath the melting curve corresponding to total and partial heats, on a mass basis. Substituting into eq 3 gives a new equation:

$$\theta_{Fi} = \theta_o - \frac{X_{2*} R_o}{\Delta L^*} \frac{1}{F_{i*}} \tag{5}$$

where ΔL^* is the molar latent heat of fusion based on the corrected total area.

Two approaches are commonly used to determine the value of K: the trial-and-error approach (Perkin-

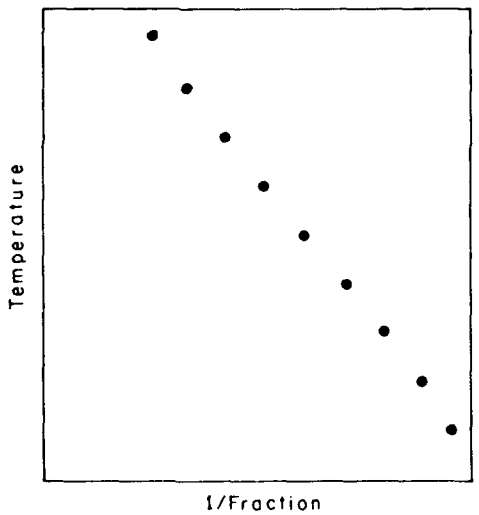

Figure 2. Fractions curve corrected for lost area.

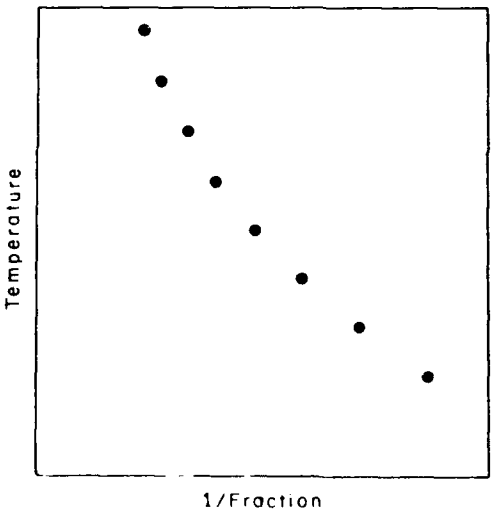

Figure 3. Fractions curve not corrected for lost area.

Elmer n.d., Plato and Glasgow 1969) and the multiple-linear-regression technique of the American Society of Testing and Materials (Wiedeman et al. 1984). The latter is obtained by adjusting eq 5 to obtain

$$A_i = -K + \theta_0 \frac{A_i}{\theta_F} + \theta_0 K - \frac{X_{2*} R \theta_{02} m}{M} \frac{1}{\theta_F} \quad (6)$$

where M is the molecular weight and m is the sample mass. Note that eq 6 contains m/M instead of the incorrect M/m of Wiedeman et al. (1984). Applying multiple linear regression to data in the form of eq 6 produces values for K, θ_0 and X_{2*}. The results of this process show how the uncorrected data of Figure 3 are adjusted to obtain the information in Figure 2.

The data analyzed by the program IMPURE, discussed below and listed in Appendix A, first applies a multiple linear regression to eq 6 to obtain values of K, θ_0 and X_{2*} and then uses the calculated value of K in eq 5 to obtain other values for θ_0 and X_{2*}. Both methods should give similar results.

DIFFERENTIAL SCANNING CALORIMETRY

There are several thermal analysis techniques available that measure changes in physical and chemical properties of materials as a function of time and temperature. By far the most informative for determining chemical purity is differential scanning calorimetry (DSC). In DSC the differential heat flow into or out of a sample (i.e. endothermic or exothermic), as compared with a reference, is measured as the temperatures of the sample and reference are changed. A typical DSC apparatus contains two test cells: one cell holds a reference, usually an empty sample container, and the other cell holds the sample in its container.

A common method of operating a DSC apparatus is the two-cycle mode, such as used by the Perkin-Elmer DSC-4. The cycling begins with both cells in thermal equilibrium. For the case of heating, the first cycle consists of constantly measuring the temperatures of both cells, averaging these temperatures, then adding equal amounts of heat to both cells to bring their average temperature to the desired value. The second cycle then involves the application of enough heat to the cell with the lowest temperature to bring the two cells into thermal equilibrium again.

The DSC apparatus is programmed to run through a desired linear temperature ramp. The programming involves choosing a starting and ending temperature as well as a heating and cooling rate. Again, for the case of heating, the run begins with both cells in thermal equilibrium at the desired starting temperature. The two-cycle process then starts with the temperature at the end of each cycling step determined by the time since the start multiplied by the heating rate plus the starting temperature. This ramping of temperature continues until enough time has elapsed to reach the desired ending temperature.

The sole output of the DSC apparatus is a voltage whose magnitude is proportional to the thermal power added in the second cycle and whose sign denotes the direction of flow. The temperature is inferred by keeping track of the elasped time, the starting temperature and the heating or cooling rates. Since the DSC output is thermal power dH/dt, the integrated output with time is therefore the total amoung of heat added to or removed from the sample $\Delta H = \int (dH/dt)\, dt$. This allows the heats of transition and the specific heats to be measured directly.

DSC calibration and operation

Since the equipment used in DSC is not exact, the temperature and energy output should always be calibrated. This is commonly done by measuring the melting characteristics of a highly pure standard such as indium and making suitable adjustments to the hardware or software. These adjustments are performed by the computer program IMPURE, listed in Appendix A, based on the operator's subjective decisions.

The energy output for the DSC is calibrated by finding a conversion factor that converts the measured fusion peak area to the known heat of fusion, taking into account the device setting. A typical fusion curve for indium is presented in Figure 4. The area associated with the latent heat of fusion is that area that begins and ends where there is significant deviation from baseline conditions. The ordinate units are given in dimensionless A/D units. These values are output from the analog-to-digital conversion of DSC voltage, which make them directly proportional to the differential thermal energy to the specimen. The abscissa units are in kelvins, previously determined from the time elapsed since the start of the run, the initial temperature and the heating rate. (The area is actually calculated with the abscissa in time units, but the temperature scale is visually superior.)

The area is determined in several steps by the operator and the program IMPURE. First the apparent baselines before and after the melting curve are determined by fitting lines through segments judged by the operator to be part of the "true" baseline. The operator then judges where the melting curve begins and ends by choosing the points where significant deviation from the two lines occur. Last, IMPURE calculates the line connecting these points and then calculates the area based on the difference between the data and the calculated line. These lines and reference points are shown Figure 5.

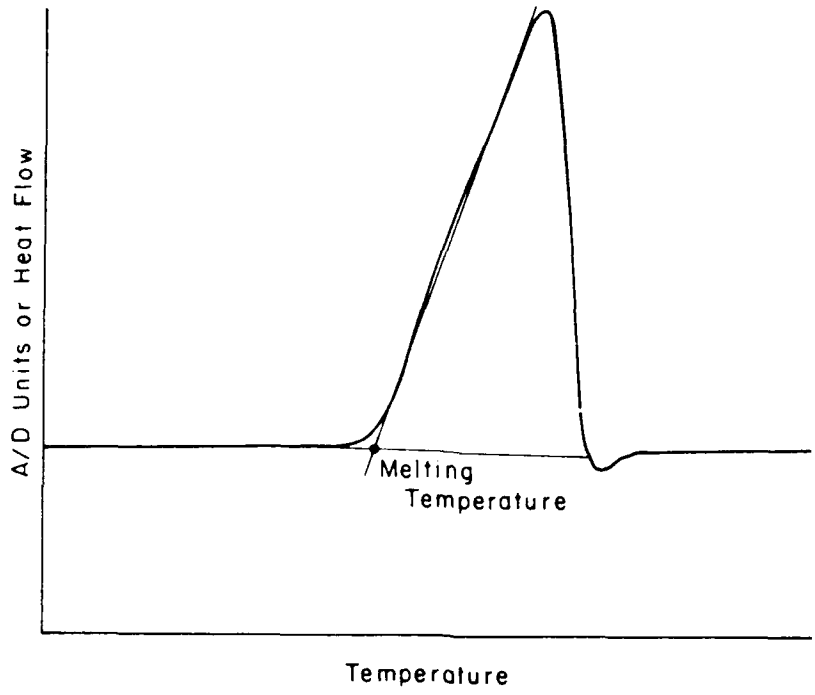

Figure 4. Typical melting temperature determination for indium.

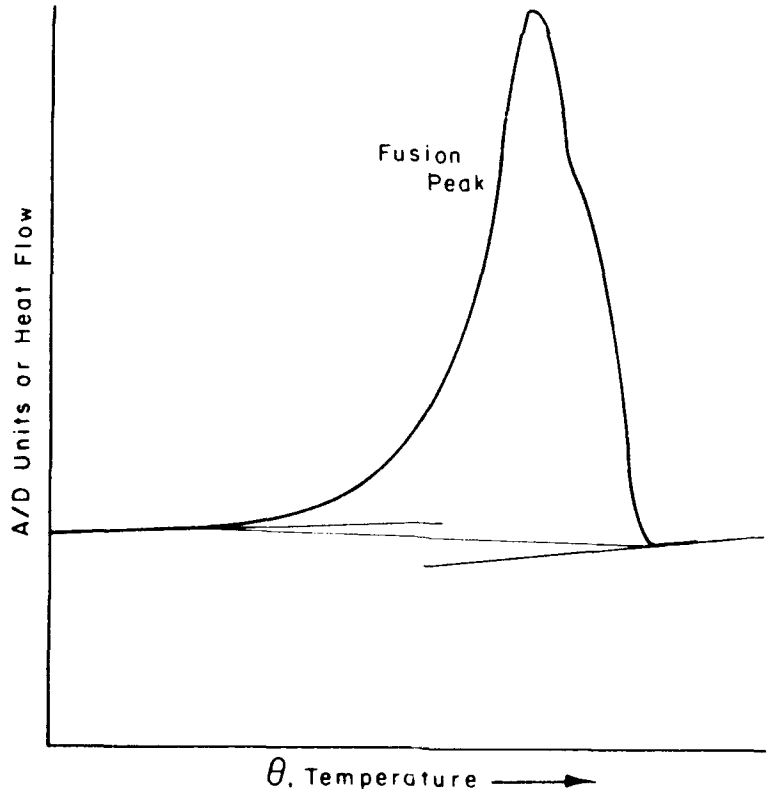

Figure 5. Baseline determination of a melting curve

IMPURE calculates the area by performing a "running" integral of the data (A/D units) minus the baseline using the 1/3 Simpson's rule (ASYST 1987). This method assumes data that are equally spaced by unit increments. IMPURE scales the result by multiplying by the time increment between data points. The resulting area (A/D units multiplied by time) should be independent of the heating rate.

The area of the fusion peak will be directly proportional to the energy of the change per unit mass ΔH (mcal/mg) and the sample mass m (mg). It will be inversely proportional to the range setting of the DSC. The range is the device's adjustable maximum power output (Perkin-Elmer 1984). Thus, the heat of fusion, on a mass basis, for a compound is

$$\Delta H = \frac{C\, A_{tot}\, R_g}{m} \tag{7}$$

where R_g is range (mcal/s) and C is the conversion factor (inverse A/D). IMPURE solves eq 7 for C from data obtained during the melting of indium with $\Delta H = 6.80$ mcal/mg (ASTM 1985).

The required temperature adjustment for the DSC is determined by measuring the difference between the melting temperature of indium determined from experimental data and its known value, 429.78 K (ASTM 1985). The melting temperature from the data is taken to be the intersection of the linear rise region of the indium melting curve (subjectively determined by the operator) with the baseline (Fig. 4). IMPURE first calculates the temperature at this intersection and then calculates the required temperature adjustment.

IMPURE stores the temperature adjustment constant and the conversion factor C for future reference. With these characteristic parameters of the DSC determined, purity measurements are then conducted.

Purity determination from DSC data

In addition to determining the total area of the fusion curve, partial areas must be calculated to use in eq 5 and 6. The operator subjectively determines the zone on the melting curve between approximately 50% melt and just before the curve decreases from linearity. This zone roughly corresponds to the linear portion of the left side of the melting curve. IMPURE then subdivides this zone into equal temperature increments and calculates these subareas.

Equation 6 is solved by multiple linear regression (ASYST 1987) for these partial areas, and the corresponding temperatures and optimal values for K, T_0 and X_2 are determined. K is then used in eq 5 to check for any possible discrepancies. Note that eq 5 and 6 use molar latent heat ΔL and not energy change per unit mass ΔH during the transition. The molar latent heat of fusion is obtained from the data by

$$\Delta L = \Delta H\, M = \frac{\left(A_{tot} + K\right) C\, R_g}{m} M \,. \tag{8}$$

Table 1. Optimum sample parameters determined by Hunter and Blaine (1984).

Specimen size	1.7 mg
Heating rate	0.5 °C/min
Data acquisition rate	120 data point/ °C

Final consideration must be given to the accuracy and precision of DSC measurements. ASTM standards proposed by Hunter and Blaine (1984) recommend optimum values of sample mass, heating rate and sampling rate for computer acquisition of impurity data. These conditions for an impurity greater than 2 mol% are presented in Table 1. In addition to determining the optimum parameters, they made two important observations. First, deviations from the optimum values result in underestimating the impurity level. Second, slight variations from these values may be tolerated but the inaccuracies thus generated are cumulative. If deviations from optimum must occur, the "true" impurity will be larger than that calculated, and care must be taken to maintain the remaining parameters at their optimum if any one parameter must deviate.

IMPURE

The computer program IMPURE was written using version 2.0 of the ASYST software package. ASYST version 2.0 is for use with the IBM AT and compatibles equipped with a math coprocessor and at least 640 K of memory. A commented listing of the IMPURE program is included in Appendix A. The function of the program is to collect thermal data from the Perkin-Elmer DSC-4 and use this collected data to determine the purity of the SARM being analyzed. This section explains the procedures used to analyze each SARM, including the operation of the IMPURE program.

Preparation of sample

Weigh a sample of 1–3 mg to an accuracy of 0.01 mg into an aluminum crucible (Perkin-Elmer Sample Kit No. 219-0062). Hermetically seal the crucible to prevent the loss of mass during the heating process.

DSC calibration

An indium sample must first be used to calibrate the DSC to determine the temperature correction before

Table 2. Molecular weights and starting and ending temperatures for seven munition standards needed for IMPURE.

Compound	Molecular weight	Starting temperature (K)	Starting temperature (°C)	Ending temperature (K)	Ending temperature (°C)
Indium	114.82	421.16	148	431.16	158
2,6 DNT	182.14	320.16	47	340.16	67
2,4 DNT	182.14	328.16	55	348.16	75
TNT	227.13	343.16	70	358.16	85
2,4,6 TNBA	241.12	380.16	107	395.16	122
Picric acid	229	385.16	112	400.16	127
1,3,5 TNB	213.11	385.16	112	400.16	127
Tetryl	287.15	395.16	122	410.16	137

each run. The indium sample is also prepared using the method described above. Place the indium sample in the left sample holder of the DSC. In the right sample holder in the DSC, place an empty sealed crucible to serve as a reference pan.

DSC setup

Use the keypad on the DSC front panel to set the minimum and maximum temperatures, the heat rate and the cooling rate. See Table 2 for values to use. The same procedure is used to set up the DSC for SARM analysis as for calibration.

Using the ASYST program IMPURE.DMO

From the DOS prompt, type "IMPURE" and press the "RETURN" key. The ASYST title screen should appear after a few moments.

Follow the instructions until the ASYST "OK" prompt is displayed. Type "LOAD IMPURE.DMO" and press "RETURN." The "OK" prompt will be redisplayed when the program has finished loading. Type "GO" to begin the program.

Input your name or initials and be sure the data disk is in Drive A. A menu should now be displayed at the bottom of the display screen.

Select "<F1> DSC" from the menu and enter the appropriate values at the prompts. When all values have been entered, select "<F3> HEAT" from the displayed menu.

Start the strip chart recorder attached to the DSC. When the temperature in the DSC has stabilized, enter a "0" (zero) in response to the prompt "GAIN (0 = 1X, 1 = 2X ...)." The computer will obtain some initial baseline data and then trigger the DSC to begin the heating process according to the parameters entered.

The length of time required to complete one heating cycle can be determined by dividing the difference between the maximum and minimum temperatures by the heat rate, e.g. (158 C to 148 C)/2.5 C per min = 4 min.

When the menu is redisplayed, select the "<F2> COOL" option to prepare the DSC for the next sample. Remove the sample from the left sample holder and discard it. Select "<F1> MAIN MENU" to return to the Main Menu. Now select "<F2> FILE" and then "<F2> Save a data file" to save the raw data. Type in any legal DOS file name in which to save the data. Select the "<F10> Main Menu" option after the data have been saved. Another sample can now be placed into the left sample holder and the above procedure repeated.

Data analysis

Once all the samples have been scanned and all the raw data have been stored on the data disk, it is time to analyze the sample data. From the Main Menu, select "<F2> FILE," then select the "<F1> Read a data file" option.

Retrieve the indium sample data by specifying the file name used to store these data earlier. Select "<F10> Main Menu" to return to the Main Menu.

Select "<F8> Indium" for special processing of the indium data. Select the "<F3> Calculate new indium" option from this menu. A plot of the raw data will be displayed on the screen, and you will be prompted to enter the beginning and end of the graph.

Move the cursor with the left and right cursor keys, and press the "HOME" key to mark the beginning of the graph. Now move the cursor to the end of the graph and press the "HOME" and then the "DEL" key to mark the end of the graph. The data will be replotted using these new end points.

Next you will be prompted to mark the beginning and end of the initial baseline. Again using the cursor keys, move the cursor to the beginning of the initial baseline and press the "HOME" key and then move to the end of the initial baseline and press the "HOME" key followed by the "DEL" key. A line will be drawn through these two points representing the initial baseline.

Table 3. Sample summary output from IMPURE.

```
Date: 09/21/88      Time: 17:28:04.80          Operator: PBB

Experiment name: 2,6 DNT
Experiment number: 2

Range (mcal/sec):  1.000
Heating rate (C/min):  2.500
Minimum temperature (C):  331.181
Maximum temperature (C):  351.160
Sampling rate (samples/sec):  2.000
Temperature increment (K):  .021

Mass of sample (mg):  2.0900
Molecular weight (g/mol):  182.140

Total area(no correction factor):  5322.667
Maximum heat value (mcal/sec):  .027
Fraction from  338.478 (K)  to  338.644 (K)

Correction factor from M.L.R:  749.843
Correction factor/total area:  .141
Coversion factor (area cal/gm mg/area sec/mcal):  2.44672E-3
Tempurature adjustment (K):  .970
Slope of indium (AD/K):  1135.206

Total heat of fusion(cal/mole):  1294.827
Heat of fusion(cal/gm):  7.109
Melting point of pure substance from M.L.R., To (K):  338.586
Melting point of pure substance from corrected data, To (K):  338.586
Freezing point depression for corrected data: -.015

Mole % from M.L.R.:  99.992
Purity (mole %) from corrected data:  99.992
```

The program will now ask for the beginning and end of the final baseline. These points should be marked in the same manner as the beginning and end points of the initial baseline.

You must now mark where the two lines intersect the curve. Again use the cursor keys to move the cursor to the point where the initial baseline first intersects the curve, i.e. the point where the curve deviates from the initial baseline. Similarly choose the point where the final baseline intersects the curve (Fig. 4). Follow the directions given to mark the beginning and end of the baseline. An "indium correction factor" will be displayed. Press any key to continue.

Save the indium file to disk. Use a different file name than the one used to save the indium raw data. When this has been completed, press "<F10> MAIN MENU" to return to the Main Menu.

With the new indium value calculated, it is now time to analyze the sample data. Begin by selecting "<F2> FILE" from the Main Menu, then select the "<F1> Read a data file" option from the file menu. Enter the file name of the first sample data file. Return to the main menu using "<F10> Main Menu."

Select "<F3> Screen plot" to display a plot of the raw data. Then select "<F9> Adjust" to adjust the raw data using the indium correction value calculated earlier. Select "<F3> Screen plot" again and notice that the upper value on the X-axis of the plot has increased. If the value did not increase, then re-read the indium file and recalculate the correction factor.

Select "<F4> Lines" and follow the instructions given. The steps that follow are very similar to the procedures used to calculate the indium standard.

When all steps have been completed, press "<F6> Summary" to print a summary to the printer (Table 3) and then press "<F5> Plotter" to send a plot of the data and the unadjusted and adjusted fraction data to the plotter (Fig. 6).

Repeat the above procedure for all samples.

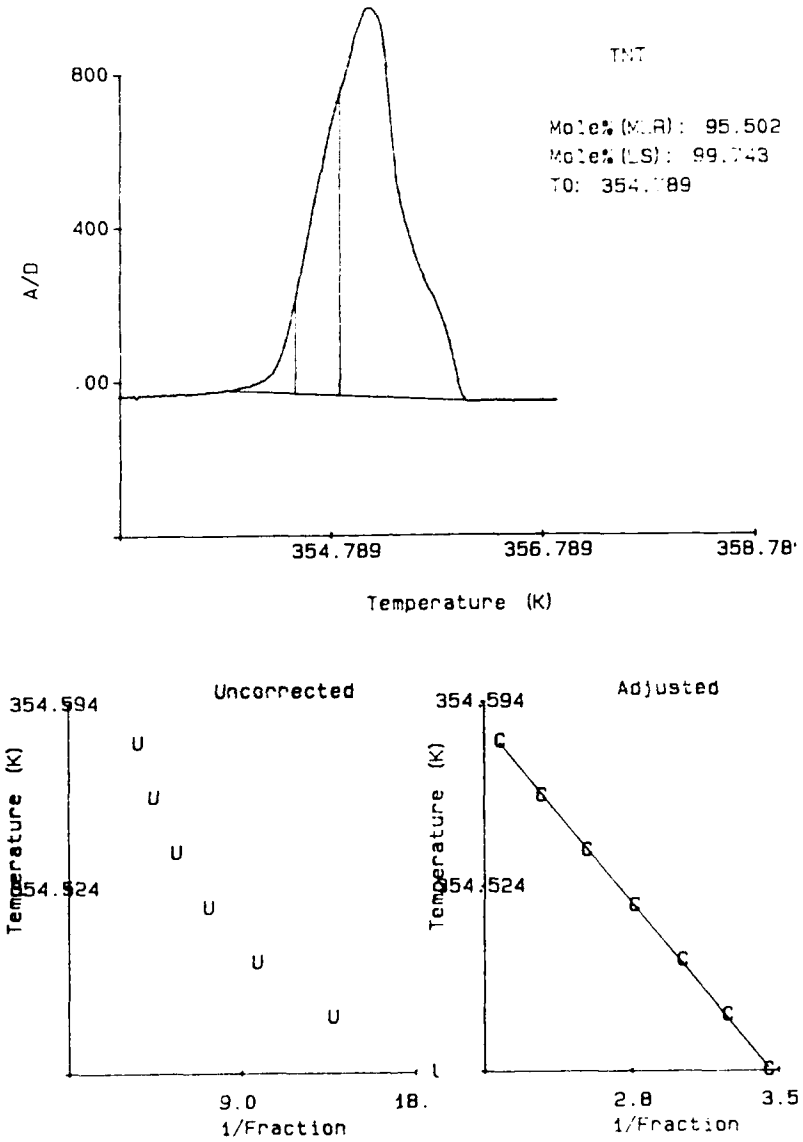

Figure 6. Sample plotting output from IMPURE.

LITERATURE CITED

American Society for Testing and Materials (1985) Standard test method for mole percent impurity by differential scanning calorimetry. E 928-85.

ASYST (1987) *ASYST User's Manual.* New York: Macmillan Publishing Company.

Gibbs, T.R. and A. Popolato (Ed.) (1980) *LASL Explosive Property Data.* Berkeley: University of California Press.

Hunter, J. and R. Blaine (1984) Optimization of accuracy and precision in the differential scanning calorimetry dynamic purity method. In *Purity Determination by Thermal Methods* (R.L. Blaine and C.K. Schoff, Ed.). ASTM Special Technical Publication 838, p. 29–38.

Jenkins, T.F., P.W. Schumacher, M.E. Walsh and C.F. Bauer (1988) Development of an analytical method for the determination of explosive residues in soil. USA Cold Regions Research and Engineering Laboratory, CRREL Report 88-8.

Meyer, R. (1987) *Explosives.* Third edition. New York: Weinheim Press.

Perkin-Elmer (1984) Instructions, Model DSC-4. Norwalk, Connecticut: Perkin-Elmer Corporation.

Perkin-Elmer (n.d.) Purity determination by differential scanning calorimetry. *Thermal Analysis Newsletter*, No. 6, Perkin-Elmer Corporation, Norwalk, Connecticut.

Plato, C. and A. Glasgow (1969) Differential scanning calorimetry as a general method for determining the purity and heat of fusion of high-purity organic chemicals. Application to 95 Compounds. *Analytical Chemistry*, **41:** 330–336.

Wentsel, R.S., R.G. Hyde, W.E. Jones, III, M.J. Wilkinson, W.E. Harward, III and J.F. Kitchens (1979) Problem definition study on 1,3-dinitrobenzene, 1,3,5-trinitrobenzene and DI-N-propyl adiate. EPA Contract no. DAMD17-77-C-7057.

Widmann, G. and H. Sommerauer (1988) Application of DSC purity analysis. *American Laboratory*, 106–112.

Wiedeman, H., R. Riesen and G. Bayer (1984) Purity determination by simultaneous differential scanning calorimetry-thermomicroscopy. In *Purity Determination by Thermal Methods* (R.L. Blaine and C.K. Schoff, Ed). ASTM Special Technical Publication 838, p.107–120.

APPENDIX A: LISTING OF THE PROGRAM IMPURE

The main routine, titled IMPURE.DMO, controls the operation of the remaining four routines: CALIBRATE.DMO, CURSOR.DMO, DSC.DMO and PLOTPRIN.DMO.

```
IMPURE.DMO

: IT!
;

: SET.PRINTER
    OUT>PRINTER
    27 EMIT  38 EMIT  108 EMIT  48 EMIT  69 EMIT  27 EMIT  38 EMIT  108 EMIT
    54 EMIT  54 EMIT  70 EMIT  27 EMIT  38 EMIT  97 EMIT  54 EMIT  76 EMIT
    CONSOLE
;

: RESET.PRINTER
    OUT>PRINTER  27 EMIT  69 EMIT  CONSOLE
;


SERIES.500
\   This program will collect analog data through the KEITHLEY Series 500
\   data collector, then plot and save the data for future use.

ECHO.OFF

INTEGER SCALAR COUNTER
        SCALAR #DATA                    \ # data actually collected
        SCALAR MAXSIZE                  \ # maximum amount of data possible
        SCALAR GAINS
        scalar upper
        SCALAR ALL                      \ # data to collect
        SCALAR  GMIN
        SCALAR  GMAX
        SCALAR  OTHER.INC

REAL    SCALAR READ.RATE                \ Sampling rate in milliseconds
        SCALAR TMIN
        SCALAR TMAX
        SCALAR TACT
        SCALAR TCUT
        SCALAR HRATE          \ Heating rate (K/min)
        SCALAR CRATE          \ Colling rate (K/min)
        SCALAR RANGE          \ Full scale range (mcal/sec)
        SCALAR XINC           \ Time for each A/D sample (min)
        SCALAR XMAX
        scalar base           \ Average of isothermal baseline
        scalar mass           \ Sample mass (mg)
        scalar mole           \ Sample molecular weight (g/mol)
        scalar M             \ slope of line at bottom of fusion curve
        scalar B             \ intercept of line at bottom of fusion curve
        scalar tot.area      \ integrated area under fusion curve
        scalar C             \ Correction factor from M.L.R
        scalar T0            \ Fusion temperature (K) from M.L.R.
        scalar x2            \ Purity from M.L.R.
        scalar R             \ Gas constant
        scalar T0INT         \ Fusion temperature (K) from adjusted data
        scalar FPD           \ Freezing point depression from adjusted data
        scalar convert       \ Area units to calories
        scalar indm          \ Slope of melting line for indium
        scalar ind0          \ Temperature adjustment to make To = 429.78 (K)

integer DIM[ 2 ] ARRAY LINE.POSITION            \ Buffer for plot positions

25 STRING OP.NAME                       \ Operators name
60 STRING TITLE                         \ Experiment title
25 STRING EXP#                          \ Experiment number
60 STRING XLBL                          \ X-axis label
60 STRING YLBL                          \ Y-axis label
```

```
14 STRING FILENAME                          \ Name of data file for storage
15 STRING LOG.DIR                           \ Directory of data diles
 2 STRING DSCF                              \ signals the use of the DSC
 2 STRING HEATF
 2 STRING COOLF

2500 MAXSIZE :=                             \ Maximum number of data allowed

 TOKEN VOLTS         \ Data array for the collection buffer
 TOKEN TEMPS         \ Temperatures (K)
 token fractions     \ Integrated areas of sub-intervals
 token grunt
 token HOLDIT        \ General purpose TOKEN

\ Windows
     23 0 24 79  WINDOW {BOTLINE}
     1 0 11 19 WINDOW {SPECS}

: LOG.DIR.SET
    LOG.DIR 1 1 "SUB  " :*.DAT"  "CAT  LOG.DIR ":=
;

: OPTIONS.MENU
     {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
 ." <F1> DSC  <F2> FILE  <F3> Screen plot  <F4> Lines  <F5> Plotter"
CR ." <F6> Summary  <F7> Data  <F8> Indium  <F9> Adjust  <F10> BYE"

;

: FILE.MENU
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
 ."  <F1> Read a data file    <F2> Save a data file    <F10> Main Menu "
;


: WRITE.FILE
\ Word for the storage of raw data
    load.overlay datafile.sov
    FILE.TEMPLATE
              3 COMMENTS
              INTEGER DIM[ 2 ] SUBFILE
              real dim[ 6 ] subfile
              REAL DIM[ #DATA ] SUBFILE
              REAL DIM[ #DATA ] SUBFILE
       END

       {BOTLINE} SCREEN.CLEAR HOME
       ." Opening file " FILENAME "TYPE
       13 EMIT FILENAME DEFER> FILE.CREATE
       {BOTLINE} SCREEN.CLEAR HOME
       ." Writting file " FILENAME "TYPE
       13 EMIT FILENAME DEFER> FILE.OPEN

              TITLE      1    >COMMENT
              EXP#       2    >COMMENT
              OP.NAME    3    >COMMENT

              2 integer ramp equiv> holdit
              2 HOLDIT [ 2 ] :=                    \ # of sets of data
              #DATA HOLDIT [ 1 ] :=                \ # of data
              1   SUBFILE HOLDIT     ARRAY>FILE

              6 real ramp equiv> holdit
              mass holdit [ 1 ] :=
              mole holdit [ 2 ] :=
              hrate holdit [ 3 ] :=
```

```
                range holdit [ 4 ] :=
                xinc holdit [ 5 ] :=
                base holdit [ 6 ] :=
                2   SUBFILE HOLDIT      ARRAY>FILE

                3   SUBFILE TEMPS       ARRAY>FILE
                4   SUBFILE VOLTS       ARRAY>FILE

        FILE.CLOSE

   ONERR: {BOTLINE} SCREEN.CLEAR HOME ." Can't open file for writting. "
                        ." Press any key to continue."
                           BELL PCKEY DROP
                           ?FILE.OPEN IF FILE.CLOSE THEN

;



: READ.FILE.IN
    clear.tokens
       " Y" DSCF ":=
       {BOTLINE} SCREEN.CLEAR HOME
        load.overlay datafile.sov
       ." Opening file " FILENAME "TYPE
       13 EMIT FILENAME DEFER> FILE.OPEN

                1 COMMENT> TITLE ":=
                2 COMMENT> EXP# ":=
                3 COMMENT> OP.NAME ":=

                 2 integer ramp equiv> holdit
                1 SUBFILE HOLDIT FILE>ARRAY

                HOLDIT [ 1 ] #DATA :=
                #DATA real RAMP EQUIV> TEMPS
                #DATA real RAMP EQUIV> VOLTS

                 6 real ramp equiv> holdit
                 2 subfile holdit file>array
                 holdit [ 1 ] mass :=
                 holdit [ 2 ] mole :=
                 holdit [ 3 ] hrate :=
                 holdit [ 4 ] range :=
                 holdit [ 5 ] xinc :=
                 holdit [ 6 ] base :=

                3   SUBFILE TEMPS       FILE>ARRAY
                4   SUBFILE VOLTS       FILE>ARRAY
       FILE.CLOSE

   ONERR: {BOTLINE} SCREEN.CLEAR HOME ." Can't open file for reading. "
                        ." Press any key to continue."
                           BELL PCKEY DROP
                           ?FILE.OPEN IF FILE.CLOSE THEN

;


: GET.FILENAME
   " DOODY.DUM" FILENAME ":=  ." Filename please ....... " "INPUT
  FILENAME ":=  13 EMIT
;

: OUT.FILE
```

```
  {DEF} SCREEN.CLEAR HOME 20 SPACES
  ." SAVING DATA...." BELL CR CR CR CR
\  LOG.DIR DEFER> DIR CR CR
  {BOTLINE} ." OUTPUT " SCREEN.CLEAR
  GET.FILENAME
;
: output.data.file
    out.file
    write.file
    escape
;

: IN.FILE
  {DEF} SCREEN.CLEAR HOME 20 SPACES
  ." READING DATA...." BELL CR CR CR CR
\  LOG.DIR DEFER> DIR CR CR
  {BOTLINE} ." INPUT " SCREEN.CLEAR
  GET.FILENAME
;

: input.data.file
    in.file
  READ.FILE.IN
  TEMPS [ 1 ] TEMPS [ #DATA ] MIN TMIN :=
  TEMPS [ 1 ] TEMPS [ #DATA ] MAX TMAX :=
    temps 1.0 * equiv> temps
    volts 1.0 * equiv> volts
  {DEF} SCREEN.CLEAR
  ESCAPE
;

: INITIALIZE
    clear.tokens
    " N" DSCF ":= " N" COOLF ":= " N" HEATF ":=
    7 ascii" symbol
\    14 AXIS.COLOR  15 LABEL.COLOR
    15 VUPORT.COLOR 0 AXIS.COLOR 0 LABEL.COLOR 15 CURSOR.COLOR 0 COLOR
    1.9872 R :=      \ cal/(deg-mol)
    -1 3 fix.format
     GRAPHICS.DISPLAY  {DEF} HOME 20 SPACES
     ." CRREL Soil Physics Purity program" CR CR CR CR
     15 SPACES ." Your name (or initials) please? "
     "INPUT OP.NAME ":=
     " A" LOG.DIR ":= CR CR CR CR 10 SPACES
     ." Make sure data disk is in drive A !!!! "
    cr ." <ret> to continue........."
     bell bell
    #input
     LOG.DIR.SET
     SCREEN.CLEAR
;
: INPUT.A.FILE
    INPUT.DATA.FILE  FILE.MENU
;

: OUTPUT.A.FILE
    OUTPUT.DATA.FILE  FILE.MENU
;

: FILE.KEYS
        F1 FUNCTION.KEY.DOES INPUT.A.FILE
        F2 FUNCTION.KEY.DOES OUTPUT.A.FILE
        F3 FUNCTION.KEY.DOES NOP
        F4 FUNCTION.KEY.DOES NOP
        F5 FUNCTION.KEY.DOES NOP
```

```
        F6 FUNCTION.KEY.DOES NOP
        F7 FUNCTION.KEY.DOES NOP
        F8 FUNCTION.KEY.DOES NOP
        F9 FUNCTION.KEY.DOES NOP
        F10 FUNCTION.KEY.DOES ESCAPE
;

: FILE.EM
        STORE.FUNCTION.KEYS
        FILE.MENU  FILE.KEYS  INTERPRET.KEYS
    ONESCAPE: RESTORE.FUNCTION.KEYS OPTIONS.MENU
;

: END.IT.ALL
    RESET.PRINTER  BYE
;
 CREATE.OVERLAY CURSOR.DMO
 CREATE.OVERLAY DSC.DMO

: LOAD.DSC
    clear.tokens
    LOAD.OVERLAY DSC.AOV
    DSC
    #DATA 1 - #DATA :=
    volts SUB[ 1 , #DATA ] 1.0 * equiv> volts
    temps SUB[ 1 , #DATA ] 1.0 * equiv> temps
;


 CREATE.OVERLAY PLOTPRINT.DMO

: PLOT.VU
    0 0 VUPORT.ORIG   1 1 VUPORT.SIZE
    .025 .008  TICK.SIZE   .5   .8    TICK.JUST
    HORIZONTAL  -.5 -1.2  8 LABEL.FORMAT
    VERTICAL    -1.2 .0  6 LABEL.FORMAT
    HORIZONTAL GRID.ON  VERTICAL GRID.ON
    1 5 AXIS.DIVISIONS  HORIZONTAL LABEL.SCALE.OFF   VERTICAL LABEL.SCALE.OFF
    HORIZONTAL AXIS.FIT.Off
    VERTICAL  AXIS.FIT.Off
;


: right.half
    0.55 .2 vuport.orig
    0.45 .8 vuport.size
    HORIZONTAL AXIS.FIT.On
    VERTICAL  AXIS.FIT.On
    .025 .008  TICK.SIZE   .5   .8    TICK.JUST
    HORIZONTAL  -.5 -1.2  5 LABEL.FORMAT
    VERTICAL    -1.2 .0  8 LABEL.FORMAT
    HORIZONTAL GRID.ON  VERTICAL GRID.ON
    3 3 AXIS.DIVISIONS  HORIZONTAL LABEL.SCALE.OFF   VERTICAL LABEL.SCALE.OFF
;
: left.half
    .0 .2 vuport.orig
    .45 .8 vuport.size
    HORIZONTAL AXIS.FIT.On
    VERTICAL  AXIS.FIT.On
    .025 .008  TICK.SIZE   .5   .8    TICK.JUST
    HORIZONTAL  -.5 -1.2  5 LABEL.FORMAT
    VERTICAL    -1.2 .0  8 LABEL.FORMAT
    HORIZONTAL GRID.ON  VERTICAL GRID.ON
    3 3 AXIS.DIVISIONS  HORIZONTAL LABEL.SCALE.OFF   VERTICAL LABEL.SCALE.OFF
;
```

```
: OTHER.LINES
    load.overlay waveops.sov
    STACK.CLEAR
    line.position [ 2 ] line.position [ 1 ] - 10 / OTHER.INC :=
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME cr ." COMPUTING........."
    other.inc 3 > if
        10 upper :=
    else
        3 other.inc :=
        line.position [ 2 ] line.position [ 1 ] - 3 / upper :=
    then
    upper real ramp equiv> fractions
    upper 1 + 1 do
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME
        stack.clear
        volts sub[ gmin , other.inc I * line.position [ 1 ] gmin - + ]
        temps sub[ gmin , other.inc I * line.position [ 1 ] gmin - + ] M * B +
        -
        integrate.data
        [ []size ] xinc 60. / INV * fractions [ I ] :=
    loop
    STACK.CLEAR
    upper real ramp 0. * 1.0 +
    fractions temps sub[ line.position [ 1 ] , upper ] /
    1.0 temps sub[ line.position [ 1 ] , upper ] /
    laminate laminate trans[ 1 , 2 ] equiv> grunt
    LOAD.OVERLAY MATFIT.SOV
    grunt fractions leastsq.multilin.fit
    dup dup
    [ 1 ] -1.0 * C :=
    [ 2 ] T0 :=
    [ 3 ] X2  :=
    T0 C *  X2 - mole * convert * range * mass / R / T0 / T0 / x2 :=
    screen.clear
    vuport.clear
    right.half
    " U" symbol
    tot.area fractions /
    temps sub[ line.position [ 1 ] , upper ] xy.auto.plot
    left.half
    " C" symbol
    tot.area C + fractions C + /
    temps sub[ line.position [ 1 ] , upper ] xy.auto.plot
    STACK.CLEAR
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME cr ." COMPUTING..........."
    LOAD.OVERLAY MATFIT.SOV
    tot.area C + fractions C + /
    temps sub[ line.position [ 1 ] , upper ]
    1  LLASTSQ.POLY.FIT
    dup
    [ 1 ] fpd :=
    [ 2 ] T0INT :=

;

: CALCULATIONS
\   4 COLOR
    solid
    TEMPS [ LINE.POSITION [ 1 ] ]  VOLTS [ LINE.POSITION [ 1 ] ]
    POSITION
    TEMPS [ LINE.POSITION [ 1 ] ]
    dup
    M * B +
    DRAW.TO
    TEMPS [ LINE.POSITION [ 2 ] ]  VOLTS [ LINE.POSITION [ 2 ] ]
    POSITION
```

```
    TEMPS [ LINE.POSITION [ 2 ] ]
    dup
    M * B +
    DRAW.TO
    OTHER.LINES
;

: LOAD.CURSOR
    LOAD.OVERLAY CURSOR.AOV
    CURSOR.POSITION
;

: LOOK.IT
    SCREEN.CLEAR  PLOT.VU  {BOTLINE} SCREEN.CLEAR
    7 ascii" symbol
 \   7 COLOR
    TEMPS  VOLTS   XY.AUTO.PLOT
;


: ADDITIONAL.LINES
    0 LINE.POSITION :=   {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
    CR ." Move cursor to choice for beginning of graph and push HOME"
    CR ." followed by choice for end of graph and push HOME then Delete"
    LOAD.CURSOR  LINE.POSITION [ 1 ] GMIN :=  LINE.POSITION [ 2 ] GMAX :=
    volts sub[ gmin , gmax gmin - ] equiv> volts
    temps sub[ gmin , gmax gmin - ] equiv> temps
    look.it
    0 LINE.POSITION :=  {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
    CR ." Move cursor to beginning of initial baseline and push HOME"
    CR ." then to its end and push HOME again, followed by Delete"
    LOAD.CURSOR STACK.CLEAR
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL cr ." COMPUTING"
    LOAD.OVERLAY MATFIT.SOV
    TEMPS SUB[ LINE.POSITION [ 1 ] , LINE.POSITION [ 2 ] LINE.POSITION [ 1 ] -
]
    VOLTS SUB[ LINE.POSITION [ 1 ] , LINE.POSITION [ 2 ] LINE.POSITION [ 1 ] -
]
    1  LEASTSQ.POLY.FIT
    LOAD.OVERLAY POLY.SOV
    TEMPS   SWAP  POLY[X]
    TEMPS   SWAP  solid
\ 2 COLOR
    XY.DATA.PLOT
    temps []size
    LINE.POSITION :=  {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
    CR ." Move cursor to beginning of final baseline and push HOME"
    CR ." then to its end and push HOME again, followed by Delete"
    LOAD.CURSOR  STACK.CLEAR
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL cr ." COMPUTING....."
    LOAD.OVERLAY MATFIT.SOV
    TEMPS SUB[ LINE.POSITION [ 1 ] , LINE.POSITION [ 2 ] LINE.POSITION [ 1 ] -
]
    VOLTS SUB[ LINE.POSITION [ 1 ] , LINE.POSITION [ 2 ] LINE.POSITION [ 1 ] -
]
    1  LEASTSQ.POLY.FIT
    LOAD.OVERLAY POLY.SOV
    TEMPS   SWAP  POLY[X]
    TEMPS   SWAP  solid
\ 9 COLOR
    XY.DATA.PLOT

    0 LINE.POSITION :=
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
    CR ." Move cursor to first baseline and curve intersection and push HOME"
    CR ." then to the other intersection and push HOME again,"
```

```
    ." followed by Delete"
    LOAD.CURSOR  STACK.CLEAR
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL cr ." COMPUTING..........."
    line.position [ 1 ] gmin := line.position [ 2 ] gmax :=
    VOLTS [ gmin ] volts [ gmax ] -
    temps [ gmin ] temps [ gmax ] -
    / M :=
    VOLTS [ gmin ] temps [ gmin ] M * -
    B :=
    stack.clear
\    volts temps look.it
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL cr ." COMPUTING..........."
   temps sub[ gmin , gmax gmin - ]
    dup M * B +
    solid
\ 15 COLOR
    XY.DATA.PLOT
    stack.clear
   volts sub[ gmin , gmax gmin - 1 + ]
   temps sub[ gmin , gmax gmin - 1 + ]
    M * B +
    -
    load.overlay waveops.sov
    integrate.data
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
    [ []size ] xinc 60. / INV * tot.area :=
;

: purity.lines
    additional.lines
    0 LINE.POSITION :=
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
CR ." Move cursor to the region where fractions are calculated, (1/10,2/3)"
   CR ." then push HOME followed by Delete"
    LOAD.CURSOR  STACK.CLEAR
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL cr ." COMPUTING..........."

    CALCULATIONS

  \  10 COLOR
    OPTIONS.MENU
;

: indium.menu
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
 ."  <F1> Read a indium file    <F2> Save indium file  "
 ."  <F3> Calculate new indium "
 cr ."                        <F10> Main Menu "
;

: indium.get
    in.file
       {BOTLINE} SCREEN.CLEAR HOME
       ." Opening file " FILENAME "TYPE
        load.overlay datafile.sov
       13 EMIT FILENAME DEFER> FILE.OPEN
            3 real ramp equiv> holdit
            1 subfile holdit file>array
                holdit [ 1 ] convert :=
                holdit [ 2 ] indm :=
                holdit [ 3 ] ind0 :=
       FILE.CLOSE
       indium.menu
   ONERR: {BOTLINE} SCREEN.CLEAR HOME ." Can't open file for reading. "
                        ." Press any key to continue."
                           BELL PCKEY DROP
```

```
                    ?FILE.OPEN IF FILE.CLOSE THEN

;

: indium.sav
    out.file
    load.overlay datafile.sov
  FILE.TEMPLATE
      real dim[ 3 ] subfile
  END

       {BOTLINE} SCREEN.CLEAR HOME
       ." Opening file " FILENAME "TYPE
       13 EMIT FILENAME DEFER> FILE.CREATE
       {BOTLINE} SCREEN.CLEAR HOME
       ." Writting file " FILENAME "TYPE
       13 EMIT FILENAME DEFER> FILE.OPEN

           3 real ramp equiv> holdit
           convert holdit [ 1 ] :=
           indm holdit [ 2 ] :=
           ind0 holdit [ 3 ] :=
           1   SUBFILE HOLDIT     ARRAY>FILE

        FILE.CLOSE
   indium.menu
   ONERR: {BOTLINE} SCREEN.CLEAR HOME ." Can't open file for writting. "
                        ." Press any key to continue."
                          BELL PCKEY DROP
                          ?FILE.OPEN IF FILE.CLOSE THEN

;


: INDIUM.calc
    look.it
    additional.lines
    6.80 mass * tot.area / range / convert :=
    0 LINE.POSITION :=
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
CR ." Move cursor to the beginning and end of linear rise,"
   CR ." then push HOME followed by Delete"
    LOAD.CURSOR  STACK.CLEAR
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME cr ." COMPUTING..........."
    LOAD.OVERLAY MATFIT.SOV
temps sub[ line.position [ 1 ] , line.position [ 2 ] line.position [ 1 ] - ]
volts sub[ line.position [ 1 ] , line.position [ 2 ] line.position [ 1 ] - ]
    1  LEASTSQ.POLY.FIT
    dup
    [ 1 ] indm :=
    [ 2 ] ind0 :=
    0 LINE.POSITION :=
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
 \   temps sub[ gmin , line.position [ 2 ] gmin - ]
 \   dup
 \   indm * ind0 +
 \   8 color
 \   solid
 \   xy.data.plot
    B ind0 - indm M - /
    dup dup indm * ind0 +
\    9 color
   " X" symbol position
    429.78 swap - ind0 :=
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
```

```
    ." Indium correction..... " ind0 .
    #input

    indium.sav

;

: indium.keys

        F1 FUNCTION.KEY.DOES indium.get
        F2 FUNCTION.KEY.DOES indium.sav
        F3 FUNCTION.KEY.DOES indium.calc
        F4 FUNCTION.KEY.DOES NOP
        F5 FUNCTION.KEY.DOES NOP
        F6 FUNCTION.KEY.DOES NOP
        F7 FUNCTION.KEY.DOES NOP
        F8 FUNCTION.KEY.DOES NOP
        F9 FUNCTION.KEY.DOES NOP
        F10 FUNCTION.KEY.DOES ESCAPE
;
: indium
 indium.menu
 store.function.keys
 indium.keys
 interpret.keys
 onescape: restore.function.keys
\ 10 COLOR
  OPTIONS.MENU
;

: Adjust
    bell
    temps ind0 + temps :=
    bell
;

: LOAD.PLOTTER
    LOAD.OVERLAY PLOTPRINT.AOV
    screen.clear
    ?PLOT.ROTATED NOT
       IF plot.rotate
       THEN
    data.plot
    corrected.data
    uncorrected.data
    ibm.graphics
    graphics.display
    options.menu
;

: LOAD.SUMMARY
    LOAD.OVERLAY PLOTPRINT.AOV
    PRINT.SUMMARY
;

: LOAD.VIEW
    LOAD.OVERLAY PLOTPRINT.AOV
    VIEW.DATA
;

: look1.it
    look.it
    options.menu
;

: GO
```

```
    GRAPHICS.DISPLAY   SCREEN.CLEAR
    SET.PRINTER   INITIALIZE   OPTIONS.MENU
    F1 FUNCTION.KEY.DOES LOAD.DSC
    F2 FUNCTION.KEY.DOES FILE.EM
    F3 FUNCTION.KEY.DOES LOOK1.IT
    F4 FUNCTION.KEY.DOES purity.LINES
    F5 FUNCTION.KEY.DOES LOAD.PLOTTER
    F6 FUNCTION.KEY.DOES LOAD.SUMMARY
    F7 FUNCTION.KEY.DOES LOAD.VIEW
    F8 FUNCTION.KEY.DOES INDIUM
    F9 FUNCTION.KEY.DOES Adjust
   F10 FUNCTION.KEY.DOES END.IT.ALL
    INTERPRET.KEYS
;
```

CALIBRATE.DMO

```
: CALIBRATE.MENU
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
."  <F1> Temperature  <F2> Heat Capacity  <F3> Integrate       <F10> Main Menu
"
;

: TEMP.MENU
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
."  <F1> Recall calibration    <F2> Create calibration    <F10> Main Menu "
;

: ALT.TEMPS
    MPS [ 1 ]  MPS [ 3 ] - TCORR :=
    MPS [ 2 ] MPS [ 4 ] - TCORR / TCORR :=

    TEMPS SUB[ 1 , #DATA ] MPS [ 3 ] - TCORR * MPS [ 4 ] +
    TEMPS SUB[ 1 , #DATA ] :=
;


: RECALL.TEMP

    {DEF} SCREEN.CLEAR HOME 20 SPACES
    ." READING STANDARDS...." BELL CR CR CR CR
    LOG.MPS DEFER> DIR CR CR
    {BOTLINE} ." Input " SCREEN.CLEAR
    GET.FILENAME

    {BOTLINE} SCREEN.CLEAR HOME
    ." Opening file " FILENAME "TYPE
    13 EMIT FILENAME DEFER> FILE.OPEN

        1 SUBFILE MPS    FILE>ARRAY

    FILE.CLOSE

    ONERR: {BOTLINE} SCREEN.CLEAR HOME ." Can't open file for reading.."
                ." Press any key to continue..."
                BELL PCKEY DROP
                ?FILE.OPEN IF FILE.CLOSE THEN

;


: STORE.TEMP
    {DEF} SCREEN.CLEAR HOME BELL
    CR CR ." Measured melting temperature (K) of standard 1.... "
    #INPUT MPS [ 1 ] :=
    CR CR ." Actual melting temperature (K) of standard 1.... "
    #INPUT MPS [ 2 ] :=
    CR CR ." Measured melting temperature (K) of standard 2.... "
    #INPUT MPS [ 3 ] :=
    CR CR ." Actual melting temperature (K) of standard 2.... "
    #INPUT MPS [ 4 ] :=

    {DEF} SCREEN.CLEAR HOME 20 SPACES
    ." SAVING STANDARDS...." BELL CR CR CR CR
    LOG.MPS DEFER> DIR CR CR
    {BOTLINE} ." Output " SCREEN.CLEAR
    GET.FILENAME

    FILE.TEMPLATE
        REAL DIM[ 4 ] SUBFILE
```

```
    END

    {BOTLINE} SCREEN.CLEAR HOME
    ." Opening file " FILENAME "TYPE
    13 EMIT FILENAME DEFER> FILE.CREATE
    {BOTLINE} SCREEN.CLEAR HOME
    ." Writting file  " FILENAME "TYPE
    13 EMIT FILENAME DEFER> FILE.OPEN

        1 SUBFILE MPS ARRAY>FILE

    FILE.CLOSE

    ONERR: {BOTLINE} SCREEN.CLEAR HOME ." Can't open file for writting.."
                ." Press any key to continue..."
               BELL PCKEY DROP
               ?FILE.OPEN IF FILE.CLOSE THEN

;

: RECALL.A.TEMP
    RECALL.TEMP
    ALT.TEMPS
    TEMP.MENU
;

: STORE.A.TEMP
    STORE.TEMP
    ALT.TEMPS
    TEMP.MENU
;




: CALIBRATE.TEMP
    BEGIN
        TEMP.MENU
        PCKEY
        CASE
            59 OF RECALL.A.TEMP      ENDOF
            60 OF STORE.A.TEMP       ENDOF
            61 OF NOP                ENDOF
            62 OF NOP                ENDOF
            63 OF NOP                ENDOF
            64 OF NOP                ENDOF
            65 OF NOP                ENDOF
            66 OF NOP                ENDOF
            67 OF NOP                ENDOF
            68 OF ESCAPE             ENDOF
        ENDCASE
    AGAIN

    ONESCAPE: NOP CALIBRATE.MENU
;


: HEAT.CAPACITY
    " Heat Capacity (J/gK)" YLBL ":=
    -5.0 273.15 + TCUT :=
    100. ADEP :=
    150. ADSTN :=
    13.3025 GSTN :=
    25.3125  GEP :=
    25.2423 GSTNP :=
```

```
    #DATA 1 + 1 DO
        TEMPS [ I ] TCUT > IF
            TEMPS [ I ] 3. ** -7.0824E-7 * CpSTN :=
            TEMPS [ I ] 2. **  8.3177E-5 * CpSTN - CpSTN :=
            TEMPS [ I ] 3.1269 * 4.2161 SWAP - CpSTN + CpSTN :=
        THEN

        TEMPS [ I ] TCUT <= IF
            0.0078 TEMPS [ I ] * 2.1158 + CpSTN :=
        THEN

        TEMPS [ I ] 2. ** 3.0172E-6 CpAL :=
        TEMPS [ I ] 8.8320E-4 * 0.8837 + CpAL - CpAL :=

        GSTNP GEP - CpAL * KCp :=
        GSTN CpSTN * KCp + KCp :=
        ADSTN ADEP - KCp / KCp :=

        KCp GSTN * CpSTN * DSTN :=
        GSMP GEP - CpAL * KCp * DSMP :=
        VOLTS [ I ] ADEP - DSMP - DSMP :=

        GSTN GSMP / DSMP * DSTN / CpSTN * VOLTS [ I ] :=

    LOOP

;

: INTEGRATE.ALL.DATA
    ." A REALLY GREAT OPTION.....SOON TO APPEAR......."
;

: CALIBRATE.KEYS
        F1 FUNCTION.KEY.DOES CALIBRATE.TEMP
        F2 FUNCTION.KEY.DOES HEAT.CAPACITY
        F3 FUNCTION.KEY.DOES INTEGRATE.ALL.DATA
        F4 FUNCTION.KEY.DOES NOP
        F5 FUNCTION.KEY.DOES NOP
        F6 FUNCTION.KEY.DOES NOP
        F7 FUNCTION.KEY.DOES NOP
        F8 FUNCTION.KEY.DOES NOP
        F9 FUNCTION.KEY.DOES NOP
        F10 FUNCTION.KEY.DOES ESCAPE
;

: CALIBRATE
        STORE.FUNCTION.KEYS
        CALIBRATE.MENU
        CALIBRATE.KEYS
        INTERPRET.KEYS

    ONESCAPE: RESTORE.FUNCTION.KEYS OPTIONS.MENU
;
```

```
CURSOR.DMO

INTEGER  SCALAR   CURINC
         SCALAR   POSINC
        scalar  tempinc

: RIGHT
    STACK.CLEAR
    posinc curinc + posinc :=
    temps []size posinc <= if temps []size posinc := then
    temps [ posinc ] volts [ posinc ]
    POSITION
    STACK.CLEAR
;
: LEFT
    STACK.CLEAR
    posinc curinc - posinc :=
    posinc  1  <= if  1  posinc := then
    temps [ posinc ] volts [ posinc ]
    POSITION
    STACK.CLEAR
;
: JUMPS
    STACK.CLEAR
     {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
    CR ." CURSOR movement increment........."
    #INPUT CURINC :=
     {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
    STACK.CLEAR
;
: POS.ARRAY
    STACK.CLEAR
    posinc  LINE.POSITION [ tempinc ] :=
     8 COLOR
     {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
     CR ." X = " temps [ POSINC ] .
     ."     Y = " volts [ POSINC ] .
     ."     I = " posinc .
     tempinc 1 + tempinc :=
    STACK.CLEAR
     7 COLOR
    STACK.CLEAR
;

: CURSOR.POSITION
    STACK.CLEAR
    179 ascii" symbol
     1 POSINC :=
     1 CURINC :=
    1 tempinc :=
     temps [ 1 ] volts [ 1 ] position
     BEGIN
     PCKEY
     CASE
         77   OF   RIGHT          ENDOF
         75   OF   LEFT           ENDOF
         73   OF   JUMPS          ENDOF
         83   OF   EXIT           ENDOF
         71   OF   POS.ARRAY      ENDOF
     ENDCASE
     AGAIN
    STACK.CLEAR
;
```

```
DSC.DMO

\ 1400 KEY.DELAY :=

INTEGER SCALAR SLOT.LOCATION
9 SLOT.LOCATION :=

0 0 A/D.TEMPLATE DSC.CHNL

50 READ.RATE :=                          \ A default setting of sampling rate

: DIO.MENU
    {BOTLINE} SCREEN.CLEAR INTEN.ON HOME BELL
    ." <F1> MAIN MENU   <F2> COOL    <F3> HEAT   <F4> HOLD"
;

: SERIES.500.CONVERSION.DELAY
    8.0 * 4.77 /
    CONVERSION.DELAY
;


: DIO.OUT
    SERIES.500 SLOT.LOCATION
        K500.CMDA POKE
        SLOT.LOCATION
        K500.CMDB POKE
;



: HOLD
    6 1 DIO.OUT
    1000 MSEC.DELAY
    7 1 DIO.OUT
    DIO.MENU
;



: COOL
    NORMAL.DISPLAY
    SCREEN.CLEAR

    3 1 DIO.OUT
    500 MSEC.DELAY
    7 1 DIO.OUT
    DIO.MENU
;


: HEAT.IT
    " Y" HEATF ":=
    TMAX TMIN - HRATE / XINC * 1 + ALL :=

    ALL MAXSIZE >  IF
      BEGIN
        SCREEN.CLEAR
        {DEF} SCREEN.CLEAR
        HOME
        CR CR CR
        ." Number of data collected must be less than" MAXSIZE .
        CR CR
        ." Present sample number is" ALL .
        CR CR
        ." New sampling rate of A/D converter...(sample/second)  "
```

```
        #INPUT XINC :=
        1. XINC / 1000 * READ.RATE :=
        XINC 60. * XINC :=      \ SECONDS PER A/D TO MINUTES PER A/D

        TMAX TMIN - HRATE / XINC * 1 + ALL :=
        ALL MAXSIZE <=
      UNTIL
    THEN
;


: HEAT
    HEAT.IT
    SCREEN.CLEAR
    {BOTLINE} SCREEN.CLEAR
    0 GAINS :=
    ." Gain (0=1X, 1=2X, 2=5X, 3=10X) "
    #INPUT GAINS :=
    SYNC.ERROR.OFF

    10 integer ramp becomes> volts
    DSC.CHNL A/D.INIT
    volts template.buffer
    1000 sync.period
    dsc.chnl gains a/d.gain
    dsc.chnl a/d.init
    home ." Collecting isothermal baseline data......"
    begin
        synchronize
        a/d.in>array
        ?buffer.full
    until
    volts 8192 - 1.0 *
    []size swap []sum swap / base :=
    stack.clear
    clear.template.buffers
    all INTEGER RAMP becomes> VOLTS
    DSC.CHNL A/D.INIT
    VOLTS TEMPLATE.BUFFER
    READ.RATE
    SYNC.PERIOD

    DSC.CHNL GAINS A/D.GAIN
    DSC.CHNL A/D.INIT
    HOME ." Collecting data, type any key to stop early..............."
    5 1 DIO.OUT
    500 MSEC.DELAY
    7 1 DIO.OUT

    BEGIN
        SYNCHRONIZE
        A/D.IN>ARRAY
        ALL ?BUFFER.INDEX <=  ?KEY OR  \ Check if enough data is collected
    UNTIL

    HOLD
    ?BUFFER.INDEX 1 -  ALL :=
    CLEAR.TEMPLATE.BUFFERS
    VOLTS []SIZE #DATA :=
    volts 8192 - 1.0 * base - equiv> volts
    #DATA real RAMP tmax tmin - #data / * tmin + equiv> TEMPS
\    OUTPUT.DATA.FILE
    DIO.MENU

;
```

```
: DSC.KEYS
        F1 FUNCTION.KEY.DOES ESCAPE
        F2 FUNCTION.KEY.DOES COOL
        F3 FUNCTION.KEY.DOES HEAT
        F4 FUNCTION.KEY.DOES HOLD
        F5 FUNCTION.KEY.DOES NOP
        F6 FUNCTION.KEY.DOES NOP
        F7 FUNCTION.KEY.DOES NOP
        F8 FUNCTION.KEY.DOES NOP
        F9 FUNCTION.KEY.DOES NOP
        F10 FUNCTION.KEY.DOES NOP
;


: DSC
    " Y" DSCF ":=
    {DEF} SCREEN.CLEAR HOME BELL
     ." Title of experiment? (name for data set) " "INPUT TITLE ":= CR CR
     ." Experiment identification number? " "INPUT EXP# ":= CR CR
    " Temperature (K)" XLBL ":=
    " A/D Value" YLBL "=
    CR CR ." Range.........(mcal/sec)  "
    #INPUT RANGE :=
\    20.0 RANGE :=
\    ." Range = " RANGE .
    CR CR ." T MIN.........(Kelvin)  "
    #INPUT TMIN :=
    CR CR ." T MAX.........(Kelvin)  "
    #INPUT TMAX :=
    CR CR ." HEAT RATE.....(K/min)  "
    #INPUT HRATE :=
    CR CR ." COOL RATE.....(K/min)  "
    #INPUT CRATE :=
    CR CR ." Sampling rate of A/D converter...(sample/second)  "
    #INPUT XINC :=
    1. XINC / 1000 * READ.RATE :=
    XINC 60. * XINC :=      \ SECONDS PER A/D TO MINUTES PER A/D
    cr cr ." molecular mass (g/mol) "
    #input mole :=
    CR CR ." Mass of sample....(mg) "
    #INPUT mass :=

    SCREEN.CLEAR
    STORE.FUNCTION.KEYS
    DIO.MENU
    DSC.KEYS
    INTERPRET.KEYS

    ONESCAPE: RESTORE.FUNCTION.KEYS OPTIONS.MENU
;
```

PLOTPRIN.DMO

```
: PLOTTER.LABEL
    NORMAL.COORDS
    0.05 YLBL "LEN 2. / 0.0175 * 0.6 SWAP - POSITION
    90 LABEL.DIR
    YLBL LABEL
    XLBL "LEN 2. / 0.0175 * .6 SWAP -
        .03 POSITION 0 LABEL.DIR
    XLBL LABEL
;

: base.line.curve
    {botline} ." base.line"
    stack.clear
    temps sub[ gmin , gmax gmin - ]
    dup
    M * B +
    3 COLOR
    XY.DATA.PLOT
;

: limit.line
    {botline} ." limit lines"
    4 color
    temps [ line.position [ 1 ] ]
    dup dup
    m * b +
    position
    volts [ line.position [ 1 ] ]
    draw.to
    temps [ line.position [ 2 ] ]
    dup dup
    m * b +
    position
    volts [ line.position [ 2 ] ]
    draw.to
;
: data.key
    normal.coords
    .7  .85 POSITION 0 LABEL.DIR
    " Mole%(MLR):"
    x2 1. swap - 100. * "." "cat
    LABEL
    .7  .8 POSITION 0 LABEL.DIR
    " Mole%(LS):"
        tot.area C + convert * range * mass / mole *
        R / t0int / t0int /
        fpd * 1.0 + 100 * "." "cat
    label
    .7  .75 POSITION 0 LABEL.DIR
    " T0:" t0 "." "cat
    LABEL
;

: data.it
    -1 3 fix.format
    HP7475
    PLOTTER.DEFAULTS
    stack.clear
    7.65 10.0 PLOTTER.SIZE
    3 3 AXIS.DIVISIONS
    vertical -1.2 0 4 label.format
    horizontal -.5 -1.2 8 label.format
```

```
    HORIZONTAL AXIS.FIT.On
    VERTICAL AXIS.FIT.On
    1 AXIS.COLOR
    1 LABEL.COLOR
    .09 .5 VUPORT.ORIG
    .99 .9 VUPORT.SIZE

    SOLID

    TEMPS
    VOLTS
    2 COLOR
    XY.AUTO.PLOT
    {BOTLINE} SCREEN.CLEAR
    base.line.curve
    limit.line
    " A/D" ylbl ":=
    " Temperature (K)" xlbl ":=
    PLOTTER.LABEL
    TITLE "LEN 2. / 0.0175 * .6 SWAP - .2 +
        .975 POSITION 0 LABEL.DIR
    TITLE LABEL
    data.key
;

: data.plot
    {BOTLINE} SCREEN.CLEAR
    ." <cr> when plotter is ready ..." BELL

    PCKEY 13 <>
        IF  OPTIONS.MENU
        THEN

    data.it

    onerr:
        data.it
;

: equat.key
    normal.coords
    .7  .8 POSITION 0 LABEL.DIR
    " F.P.D. = " fpd "."
    "cat
    LABEL
    .7 .7 position 0 label.dir
    " To = " t0int "." "cat
    label
;

: corrected.it
    -1 3 fix.format

    HP7475
    PLOTTER.DEFAULTS
    7.65 10.0 PLOTTER.SIZE
    .58 .05 VUPORT.ORIG
    .5 .4 VUPORT.SIZE

    2 2 AXIS.DIVISIONS
    HORIZONTAL AXIS.FIT.ON
    VERTICAL AXIS.FIT.ON
    vertical -1.2 0 8 label.format
    horizontal -.6 -1.2 4 label.format
    1 AXIS.COLOR
    1 LABEL.COLOR
```

```
    stack.clear

\    0.0 tot.area fractions c + / []max horizontal world.set
\    temps sub[ line.position [ 1 ] , upper ]
\    []MIN/MAX VERTICAL WORLD.SET
\    XY.AXIS.PLOT
    " C" symbol
    tot.area c + fractions c + /
    temps sub[ line.position [ 1 ] , upper ]
    xy.AUTO.Plot
    3 COLOR
    solid
\    2 real ramp equiv> holdit
\    tot.area fractions c + / []max holdit [ 1 ] :=
\    0.0 holdit [ 2 ] :=
\    HOLDIT dup  fpd * toint +
    TOT.AREA C + FRACTIONS C + / DUP FPD * TOINT +
    XY.DATA.PLOT
    " Temperature (K)" ylbl ":=
    " 1/Fraction" xlbl ":=
    " Adjusted" title ":=
    PLOTTER.LABEL
    TITLE "LEN 2. / 0.0175 * .6 SWAP -
        .975 POSITION 0 LABEL.DIR
    TITLE LABEL
\    equat.key
;

: corrected.data
    corrected.it

    onerr: corrected.it
;

: uncorrected.it
    -1 3 fix.format

    HP7475
    PLOTTER.DEFAULTS
    7.65 10.0 PLOTTER.SIZE
    .09 .05 VUPORT.ORIG
    .5 .4 VUPORT.SIZE

    2 2 AXIS.DIVISIONS
    HORIZONTAL AXIS.FIT.ON
    VERTICAL AXIS.FIT.On
    vertical -1.2 0 8 label.format
    horizontal -.6 -1.2 4 label.format
    1 AXIS.COLOR
    1 LABEL.COLOR
    stack.clear

\    0.0 tot.area fractions / []max horizontal world.set
    " U" symbol
    tot.area fractions /
    temps sub[ line.position [ 1 ] , upper ]
    XY.AUTO.PLOT
    2 COLOR
    " Uncorrected" title ":=
    PLOTTER.LABEL
    TITLE "LEN 2. / 0.0175 * .6 SWAP -
        .975 POSITION 0 LABEL.DIR
    TITLE LABEL
;
: uncorrected.data
```

```
    uncorrected.it

    onerr: uncorrected.it
;

: PRINT.THE.SUMMARY
    {BOTLINE} SCREEN.CLEAR

    -1 3 fix.format
    35 spaces ." PURITY ANALYSIS" cr cr
    10 SPACES ." Date: " .DATE  5 SPACES ." Time: " .TIME
    8 SPACES ." Operator: " OP.NAME "TYPE CR CR
    10 SPACES ." Experiment name: " TITLE "TYPE CR
    10 SPACES ." Experiment number: " EXP# "TYPE CR CR

    10 SPACES ." Range (mcal/sec): " RANGE . CR
    10 SPACES ." Heating rate (C/min): " HRATE . CR
    10 SPACES ." Minimum temperature (K): " TMIN . CR
    10 SPACES ." Maximum temperature (K): " TMAX . CR
    10 SPACES ." Sampling rate (samples/sec): "  XINC 60 / . CR
    10 spaces ." Temperature increment (K): " tmax tmin - #data / . cr cr
    -1 4 fix.format
    10 spaces ." Mass of sample (mg): " mass . cr
    -1 3 fix.format
    10 spaces ." Molecular weight (g/mol): " mole . cr cr

    10 spaces ." Total area(no correction factor): " tot.area . cr
    10 spaces ." Maximum heat value (mcal/sec): "
                volts sub[ gmin , gmax gmin - ]
              []max 8192 / range * . cr
    10 spaces ." Fraction from " temps [ line.position [ 1 ] ] .
    ." (K)  to " temps [ line.position [ 2 ] ] . ." (K)"cr cr

    10 spaces ." Correction factor from M.L.R: " c . cr
    10 spaces ." Correction factor/total area: " c tot.area / . cr
    10 spaces ." Coversion factor (area cal/gm mg/area sec/mcal): "
    -1 5 SCI.FORMAT
                convert . cr
    -1 3 FIX.FORMAT
    10 spaces ." Tempurature adjustment (K): " ind0 . cr
    10 spaces ." Slope of indium (AD/K): " indm . cr cr

    10 spaces ." Total heat of fusion(cal/mole): "
          tot.area C + convert * range * mass / MOLE * . cr
    10 spaces ." Heat of fusion(cal/gm): "
          tot.area C + convert * range * mass / . cr
 10 spaces ." Melting point of pure substance from M.L.R., To (K): " T0 . cr
10 spaces ." Melting point of pure substance from corrected data, To (K): "
         t0int . cr
    10 spaces ." Freezing point depression for corrected data: " fpd . cr cr
    stack.clear
    10 spaces ." Mole % from M.L.R.: " x2 1. swap - 100. * . cr
    10 spaces ." Purity (mole %) from corrected data: "
         tot.area C + convert * range * mass / mole *
         R / t0int / t0int /
         fpd * 1.0 + 100 * . cr cr

\    XLEL "TYPE 8 SPACES YLBL "TYPE
\    CR CR 10 4 SCI.FORMAT
\    #DATA 1 + 1
\        DO
\            VOLTS [ I ]
\            TEMPS [ I ]
\
\            20 SPACES . 8 SPACES . CR
\        LOOP
```

```
    10 spaces ."
;

: PRINT.SUMMARY
    OUT>PRINTER  CONSOLE.OFF  {DEF} CR CR CR CR    \ OPEN LINE TO PRINTER
    PRINT.THE.SUMMARY
    12 EMIT
\    " A:M" FILENAME "CAT DEFER> OUT>FILE
\    CONSOLE.OFF
\    PRINT.THE.SUMMARY
\    OUT>FILE.CLOSE
    CONSOLE OPTIONS.MENU
;

: VIEW.DATA
   GRAPHICS.DISPLAY
   {DEF} SCREEN.CLEAR HOME
    -1 3 fix.format
   21 SPACES XLBL "TYPE 14 SPACES YLBL "TYPE CR
    #DATA 1 + 1
        DO
          VOLTS [ I ]
          TEMPS [ I ]
          20 SPACES . 8 SPACES . CR

          I 22 MODULO 0 = IF
             {BOTLINE} SCREEN CLEAR
             ." type any key to continue....."
             PCKEY ?DROP DROP
             {DEF} SCREEN CLEAR HOME
             21 SPACES XLBL "TYPE 14 SPACES YLBL "TYPE CR
          THEN
        LOOP
   {BOTLINE} SCREEN.CLEAR
   ." type any key to continue....."
   PCKEY ?DROP DROP
   {DEF} SCREEN.CLEAR HOME
   OPTIONS.MENU
;
```

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	August 1991	

4. TITLE AND SUBTITLE
Operators Manual for Determining Mole Percent Purity Using IMPURE

5. FUNDING NUMBERS

6. AUTHORS
Dean Pidgeon and Patrick B. Black

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Cold Regions Research and Engineering Laboratory
72 Lyme Road
Hanover, N.H. 03755-1290

8. PERFORMING ORGANIZATION REPORT NUMBER
Special Report 91-11

9. SPONSORING MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Toxic and Hazardous Materials Agency
Aberdeen Proving Ground, Maryland

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
CETHA-TE-CR-91043

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

Available from NTIS, Springfield, Virginia 22161.

12b. DISTRIBUTION CODE

13. ABSTRACT (*Maximum 200 words*)

This report presents the laboratory procedures and operation of the computer program IMPURE, which allows the operator to measure the mole percent purity of Standard Analytical Reference Materials (SARMs). Melting point temperature, freezing point depression and heat of fusion are measured by differential scanning calorimetry (DSC). These data are then used in van't Hoff's equation to determine molar purity. IMPURE was written to control the operation of the DSC and the analyses of the collected data according to American Society of Testing and Materials (ASTM) standards.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Differential scanning calorimetry Standard analytical reference materials Mole percent purity	39
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL